AD-A134 319

(12)

FINAL REPORT

Contract MDA 903-82-C-0390

BELL-NORTHERN RESEARCH LTD.
P.O. Box 3511
Station C
Ottawa, Ontario
K1Y 4H7







Prepared by
Bell-Northern Research Ltd.
Ottawa, Ontario

# CODING OF FULL MOTION COLOR VIDEO SIGNALS AT 50 KB/S

## FINAL REPORT

Contract MDA 903-82-C-0390

PREPARED BY: BELL-NORTHERN RESEARCH LTD.
P.O. Box 3511
Station C
Ottawa, Ontario
K1Y 4H7

AUTHORS: E. Gulko
S. Sabri



AUGUST 1983

PREPARED FOR: DEFENSE ADVANCED RESEARCH PROJECTS AGENCY
(DARPA) UNDER CONTRACT NO. MDA90382C0390

## Summary

This report describes a video coder operating at 56 kb/s or 64 kb/s. The coder was designed for possible use in the proposed National Command Authority Teleconferencing System (NCATS) [1]. The system provides multisite, multimedia conferencing. It assumes one participant per site. Under certain conditions, availability of communications bandwidth can be drastically lowered. This report presents the coding algorithm for operation at 56 – 64 kb/s transmission bit rates. In this case, a capacity of 50 kb/s is allocated for the video information, while the remaining 6 – 14 kb/s is reserved for framing, error control, and other overhead.

The NTSC color video signal is sampled at 14.3 MHz (four times the color subcarrier frequency) and uniformly quantized to 8 bits per sample resulting in a transmission rate of 114 Mb/s. For video conferencing applications, on the other hand, the bit rate can be significantly reduced. To attain the 50 kb/s bit rate, the bandwidth compression ratio of the order of 2000:1 is required. The reduction in the bit rate is achieved by removing the redundant information from the original signal, and exploiting the video conferencing environment.

The redundancy present in the signal falls into two main categories: statistical and perceptual. Statistical redundancy manifests itself as a high degree of correlation between adjacent picture elements (pels) and can be removed by appropriate processing techniques. The processes involved are reversible, that is, the inverse operation can be performed to reconstruct the original signal without distortions. Perceptual redundancy, on the other hand, once removed, cannot be recovered. However, the distortion introduced by removing this type of redundancy varies from imperceptible to tolerable

by the human visual perceptual system.

The present work is a continuation of the previous work [2] on the multi-bit-rate coder capable of dynamically changing the bit rate from 1.5 Mb/s down to 64 Kb/s. The transmission at the required rates is realized using interframe coding and several other bit rate reduction techniques which exploit the statistical properties of signal, the properties of the human visual system, and the video conference environment.

The video environment in the NCATS specifies a single participant per conference site. Therefore, the full frame need not be coded; instead, a window of 256 pels per line and 100 lines per field (approximately one-seventh of the screen size) is used. The size of this window is large enough to accommodate a head-and-shoulders view of the participant with sufficient resolution.

A key element to achieving the required bit rate reduction while maintaining acceptable picture quality is the utilization of motion estimation and compensation techniques [3], [4], [5]. In conventional interframe coders (without motion compensation), a prediction of the current frame picture element (pel) is formed using corresponding previous frame picture element(s). The prediction error, i.e., the difference between the current pel value and the predicted value, is quantized, processed and transmitted. Therefore, only areas of the picture that have changed from one frame to the next have to be coded and transmitted. In movement compensated coding, the displacement of different objects in the picture, i.e., participant motion from one frame to the next, is estimated. The prediction is formed in the direction of motion, using the displaced frame element as prediction. The percentage of picture area that is fully predictable (prediction error is below a preset threshold) is increased. In addition, the predition error in picture areas which are not fully predictable is significantly reduced. The final result is a significant reduction in the bit rate.

The multi-bit-rate coder gives very good picture quality at the rates 256 Kb/s and above. The quality at 64 Kb/s was also considered usable, although some distortions were noticeable. The problems manifested themselves as jerkiness in the areas of large displacements, and aliasing. Both problems are attributed to inadequate 3-D subsampling and interpolation techniques used in that coder.

The present report concentrates on two enhancements which dramatically improve the coder performance at the 56 – 64 kb/s rate. The first enhancement is the modification of the spatial subsampling pattern which reduces the aliasing in the reconstructed signal. Placement of the spatial interpolator both at the transmitter (within the feedback loop) and the receiver is also found to improve the picture quality compared to the interpolator at the receiver alone. The second enhancement, perhaps the more significant of the two, is the motion compensated temporal field interpolation. Here, instead of conventional linear techniques which inherently result in a loss of resolution, the interpolation is performed by gradual shifting of the moving objects in the direction of motion. The displacement is estimated using the algorithms similar to those utilized in motion compensated coding.

The coder has been simulated on the VAX 11/780 computer and the processed images displayed using BNR/INRS real-time display facilities. The results obtained using the new interpolation techniques were found to be far superior to those using previous algorithms. In particular, the resolution and the motion rendition are greatly improved, and the artifacts are minimized.

Table of Contents.

List of Figures.

# 1. Introduction.

## 1.1. Problem Statement and Review of the Previous Work.

This report describes a video coder operating at 56 kb/s or 64 kb/s. The coder was designed for possible use in the proposed National Command Authority Teleconferencing System (NCATS) [1]. The system provides multisite, multimedia conferencing. It assumes one participant per site. Under certain situations, availability of communications bandwidth can be drastically lowered. This report presents the coding algorithm for operation at 56 – 64 kb/s transmission bit rate. In this case, a capacity of 50 kb/s is allocated for the video information, while the remaining 6 – 14 kb/s is reserved for framing, error control, and other overhead.

The NTSC color video signal for broadcast TV, sampled at 14.3 MHz (four times the color subcarrier frequency) and uniformly quantized to 8 bits per sample results in a transmission rate of 114 Mb/s. For video conferencing applications, the bit rate can be significantly reduced. To attain the 50 kb/s bit rate, a bandwidth compression ratio of the order of 2000:1 is required. The reduction in the bit rate is achieved by removing the redundant information from the original signal, and exploiting the video conferencing environment.

The redundancy present in the signal falls into two main categories: statistical and perceptual. Statistical redundancy manifests itself as a high degree of correlation between adjacent picture elements (pels) and can be removed by appropriate processing techniques. The processes involved are, in general, reversible, that is, the inverse operation can be performed to

reconstruct the original signal without distortions. Exploitation fo perceptual redundancy results in lossy or irreversible coding. However, the distortion introduced by removing this type of redundancy varies from imperceptible to tolerable by the human visual perceptual system.

The present work is a continuation of the previous work [2] on the multi-bit-rate coder capable of dynamically changing the bit rate from 1.5 Mb/s down to 64 Kb/s. Required rates are realized using interframe coding and several other bit rate reduction techniques which exploit the statistical properties of signal, the properties of the human visual system, and the video conference environment.

The video environment in the NCATS specifies a single participant per conference site. Therefore, the full frame need not be coded; instead, a window of 256 pels per line and 100 lines per field (approximately one-seventh of the screen size) is adequate. The size of this window is large enough to accommodate a head-and-shoulders view of the participant with sufficient resolution.

A key element in achieving the required bit rate reduction while maintaining acceptable picture quality is the utilization of motion estimation and compensation techniques [3], [4], [5]. In conventional interframe coders (without motion compensation), a prediction of the current frame picture element (pel) is formed using corresponding previous frame picture element(s). The prediction error, i.e., the difference between the current pel value and the predicted value, is quantized, processed and transmitted. Therefore, only areas of the picture that have changed from one frame to the next have to be coded and transmitted. In movement compensated coding, the displacement of different objects in the picture, i.e., participant motion from one frame to the next, is estimated. The prediction is formed in the direction of motion, using the displaced frame element as prediction. The percentage of picture area that is fully predictable (prediction error is below a preset threshold)

is increased. In addition, the prediction error in picture areas which are not fully predictable is significantly reduced. The final result is a significant reduction in the bit rate.

The multi-bit-rate coder gives very good picture quality at the rates 256 Kb/s and above. The quality at 64 Kb/s was also considered usable, although some distortions were noticeable. The problems manifested themselves as jerkiness in the areas of large displacements, and aliasing, which stems from coarse spatial sampling.

The present report concentrates on two enhancements which dramatically improve the coder performance at 56 – 64 kb/s rates. The first enhancement is the modification of the spatial subsampling pattern which reduces the aliasing in the reconstructed signal. The place where the spatial interpolator is inserted in the coding path was also found to be important. The second enhancement, perhaps the more significant of the two, is the motion compensated temporal field interpolation. Here, instead of conventional linear techniques which inherently result in a loss of resolution, the interpolation is performed by gradual shifting of the moving objects in the direction of motion. The displacement is estimated using the algorithms similar to those utilized in motion compensated coding. As a result, the resolution and the motion rendition are greatly improved, and the artifacts are minimized.

## 1.2. General System Description.

Block diagrams of the transmitter and the receiver are shown in Figures 1.1 and 1.2 respectively. The coder can operate in a number of different modes, that is, the parameters in different blocks can change adaptively depending on the activity in the image. The details of the multi-mode operation have been described in [2].

The composite color NTSC signal generated from the camera is sampled at $4f_{sc}$ ($f_{sc} = 3.58MHz$, the color subcarrier frequency) and digitized using 8 bits per sample PCM. Only a center window about 1/7-th the size of the complete image is retained for further processing.

The composite digitized signal is fed into a demodulator which separates the input into three components: luminance $Y$, and chrominance $I$ and $Q$. The demodulator block also multiplexes these components into the form appropriate for the use in subsequent processes.

The input signal usually contains noise caused by the use of inexpensive cameras and/or by low lighting conditions. Noise in the video signal will reduce the efficiency of the coder, and will unnecessarily increase the generated bit rate. Therefore, the use of noise reduction techniques is highly desirable. In Figure 1.1, the noise reduction unit immediately follows the demodulator.

The function of the next block is to sub-sample the signal spatially and temporally, so that the transmission bit rate constraint can be met. The issues involved here are those of aliasing and loss of spatial and temporal

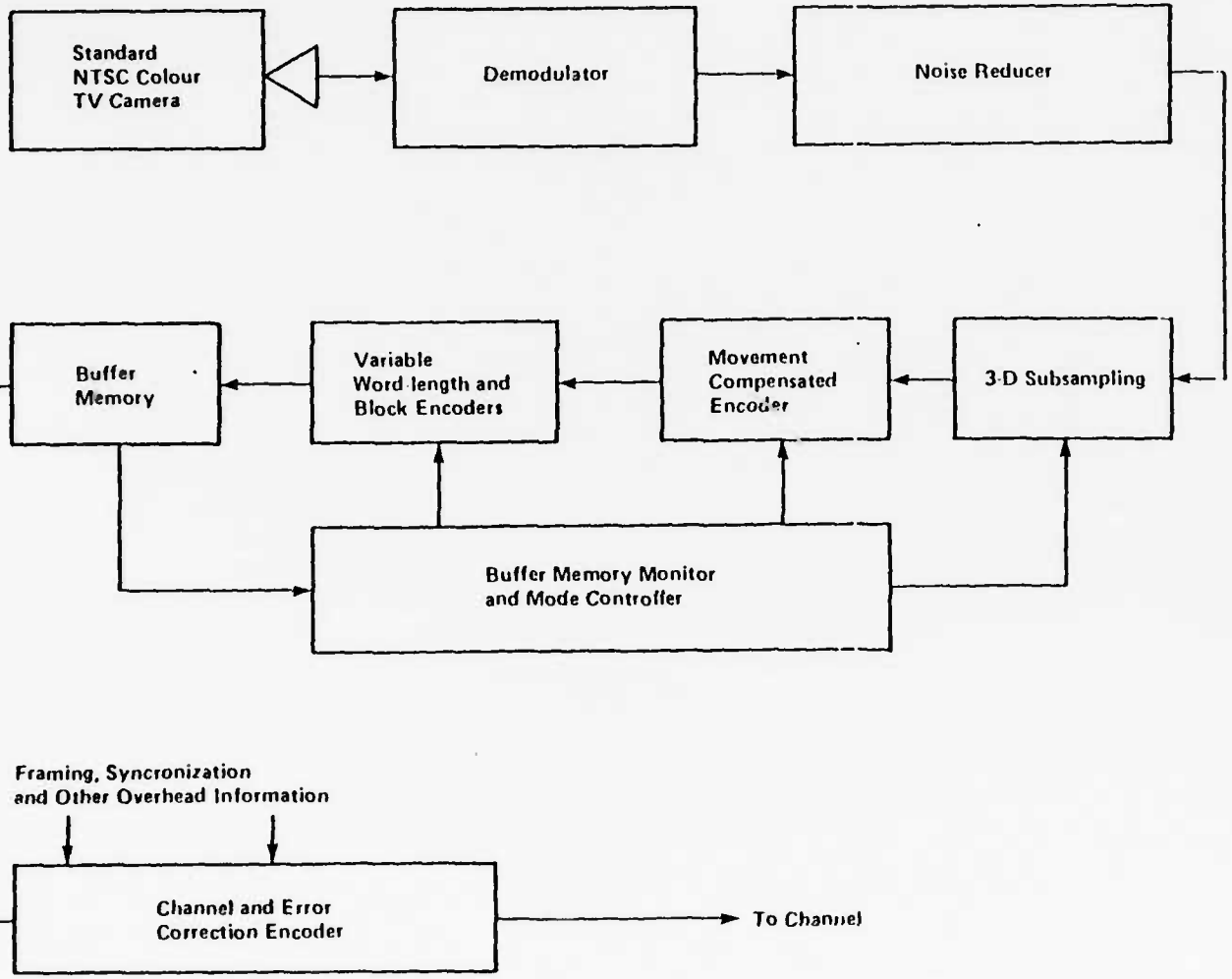



Figure 1.1: General Block Diagram of the Complete Video Encoder

resolution. To avoid spatial aliasing, the video signal is bandlimited to about 2MHz. and sampled at $1/_{4}$ using the line quincunx pattern. Further, the image sequences are subsampled temporally at a ratio varying between 4:1 to 16:1 depending on the mode of operation. The inverse operation, the motion compensated field interpolation implemented at the receiver, is the major development described in this report.

Following the sub-sampling stage is the movement compensated coder, the heart of the system. The unit estimates the displacement of different areas of the image, uses it to predict the intensities of pels in the upcoming field, and outputs the prediction errors.

Further reduction in the bit rate is derived from the fact that the prediction error distribution is non-uniform. Large regions, either stationary or translationally moving, may be completely predictable, i.e., produce a prediction error less than a preset threshold. Block coding takes advantage of this by allocating one bit per fixed-size block to indicate whether or not the block is predictable. If it is, no additional information is required. If not, variable-length encoding is used, whereby codewords of shorter length are assigned to more frequently encountered prediction error values. As a result, the average output bit rate is reduced. An unavoidable side-effect of the variable word-length encoding, however, is that its output bit rate fluctuates depending on the properties of the image being encoded. This problem is overcome using a first-in-first-out buffer.

The output buffer serves a dual purpose. First, it converts a variable rate data input stream into a constant rate output. Secondly, it incorporates the controller for the multi-mode coder. It is this controller that switches the modes of operation so as to maintain the output transmission rate at the specified level.

The last stage of the transmitter is the channel coder which protects the

digital signal against the channel errors. Finally, framing and synchronization bits are added to complete the formatting process.

The receiver performs the inverse of the processes implemented by the transmitter to reconstruct the video signal as shown in Figure 1.2.

Many functional blocks of the above system have been developed and described previously [2], and are therefore only briefly reviewed. However, the new features of the coder, such as the spatial sampling pattern and movement compensated field interpolation, will be described here in detail.

## 1.3. Report Outline.

The report is structured as follows. Chapter 2 describes the operation of the motion compensated field interpolation. Chapter 3 reviews the main functional blocks of the motion compensated coder and concentrates more specifically on the spatial sampling and interpolation. Finally, Chapter 4 presents the conclusions and directions for future work.

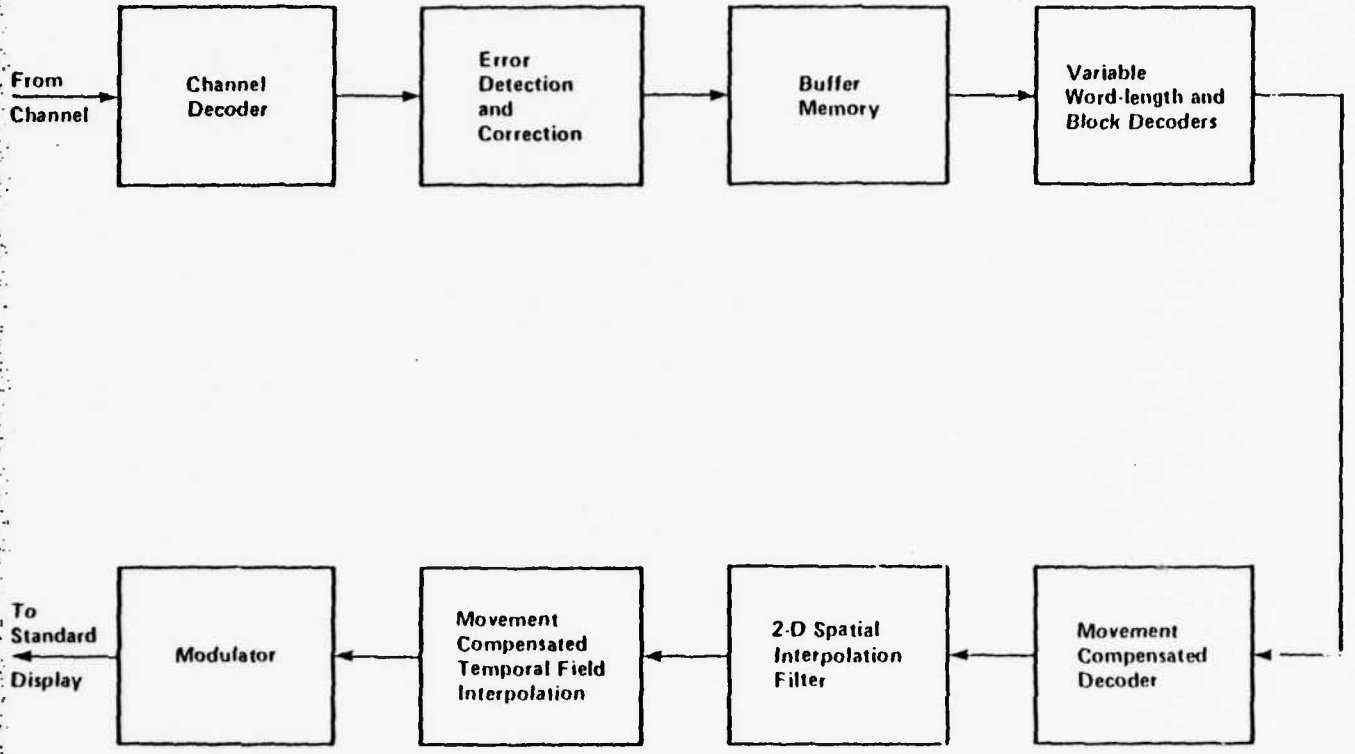



Figure 1.2: General Block Diagram of the Complete Vidio Decoder

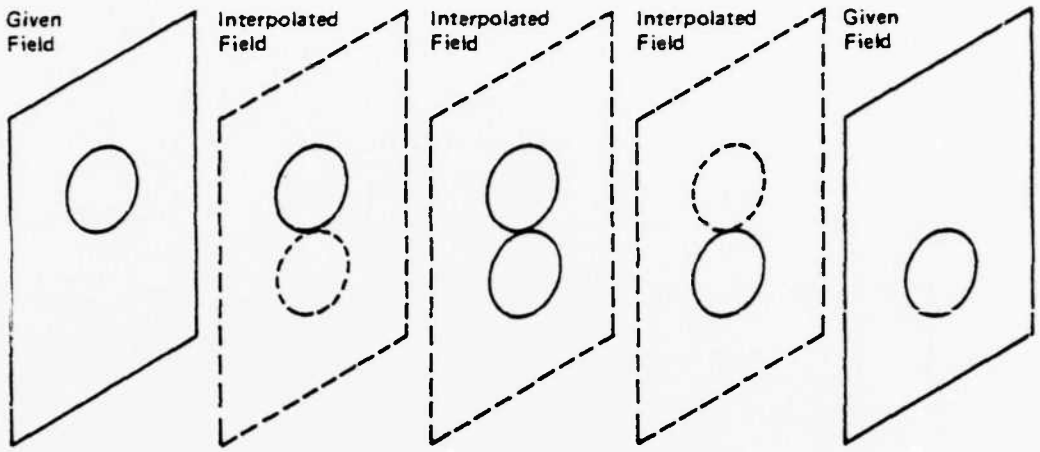
## 2. Temporal Subsampling and Motion Compensated Field Interpolation.

### 2.1. Problem Statement.

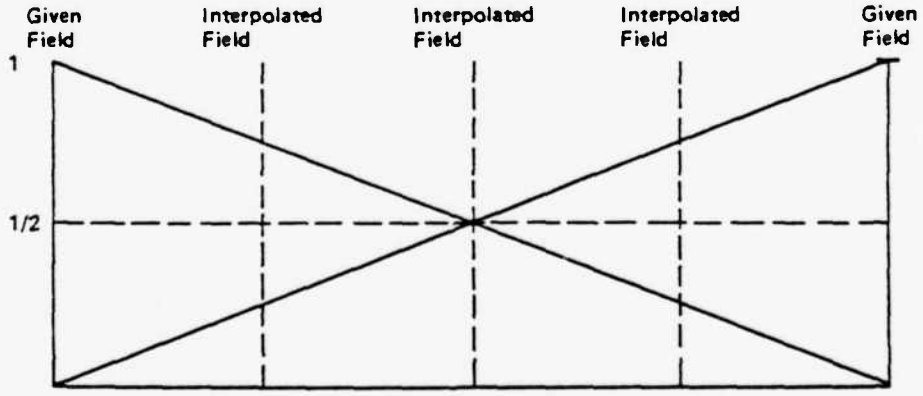
In order to achieve high compression ratios, temporal field subsampling must be used in addition to the spatial subsampling. In this case, the receiver has to reconstruct missing fields.

A few approaches exist for reconstruction of missing fields. The most rudimentary one is that of field repeat. The procedure is analogous to sample-and-hold used for one-dimensional signals. In the case of video images, this technique amounts to repeating the last transmitted field until a new field becomes available. The drawbacks of this technique are rather obvious, as the high degree of jerkiness is produced in moving areas. The picture quality is generally considered unacceptable even at small temporal subsampling ratios.

The next logical technique to consider is linear interpolation. One form this process can take is illustrated in Figure 2.1. The interpolated field, say $k$, is derived from two given fields, $i$ and $j$, one on either side of field $k$. Interpolation is performed by weighting the intensity values of the two given images $i$ and $j$ by coefficients that are inversely proportional to the distance from these fields to field $k$. The new field $k$ is obtained by adding the two weighted images. In order to avoid temporal aliasing, prefiltering is required. For typical motion in conferencing environment, the use of linear interpolation produces acceptable results for temporal subsampling ratios below 4 : 1. At 4 : 1, distortions begin to appear. The distortions take form of the loss of resolution within moving areas, appearance of double or "ghost" images, and jerkiness of fast moving objects.




(a) Linearly Interpolated Images




(b) Weighting Coefficients

Figure 2.1: Linearly Weighted Interpolation

The technique that overcomes most of these problems is the motion compensated field interpolation. Here, the motion of different objects in the picture is first estimated, as in the case of motion compensated coding. Then in the interpolated field, the objects will appear displaced proportionately in the direction of motion. In the ideal case, that is, when the motion is translational, and the displacement estimates are correct, this method gives zero distortion, except in the newly exposed regions. As in the linear interpolation case above, the motion compensated method operates only on two consecutively subsampled fields, henceforth referred to as current and previous, without regard to the further past or future. The algorithm can be divided into two parts: i) motion estimation and ii) field interpolation.

The motion estimation algorithm establishes a map from the pels in the current field onto the previous field. This map, also called a displacement field, consists of vectors originating at a pel in the current field, and pointing to the corresponding pel in the previous field. The intention is to trace the motion of individual pels between the two fields.

The interpolation part of the algorithm amounts to picking the intensity values of the pels on the ends of each displacement vector, multiplying each by a complementary weighting coefficient, adding the weighted values, and inserting the result at the intersection of the interpolated field and the displacement vector. Weighting coefficients are similar to ones used in linear interpolation mentioned earlier. Possible problems that arise in this process and their solution will be treated later. The concept of movement compensated interpolation is illustrated in Figure 2.2, and its implementation is shown in Figure 2.3.

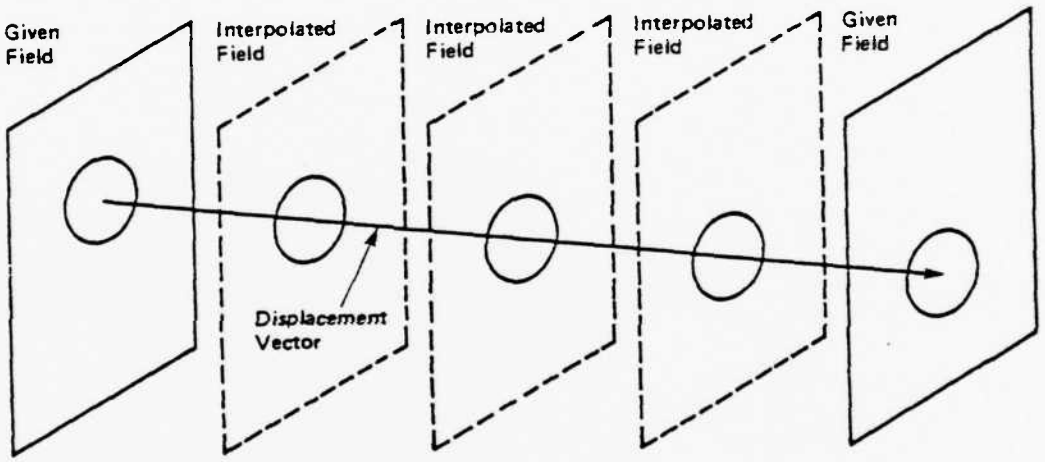



Figure 2.2: Movement Compensated Field Interpolation

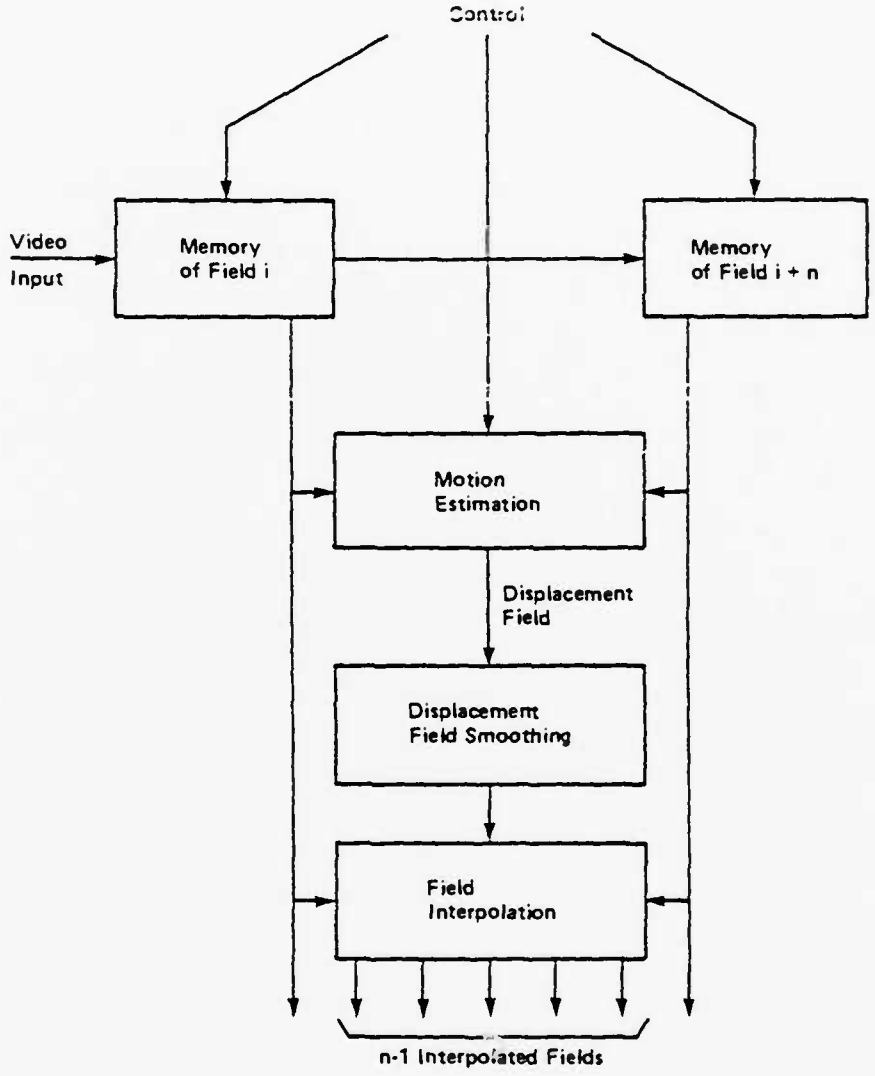



Figure 2.3: The Block Diagram of n:1
Temporal Field Interpolation

## 2.2. Motion Estimation and Field Interpolation.

We begin the description of the algorithm by defining more rigorously the displacement vector. In the previous section, we stated that a displacement vector is one which connects two corresponding pels in consecutively sub-sampled fields. In general terms, corresponding pels are pels which coincide with the same point of an object depicted in two fields, one image possibly displaced from the other. In order to perform mathematical manipulations, we will assume an image model where the corresponding pels have the closest match in intensity. The displacement will then be defined as follows.

Let $I(x, y, n)$ be the luminance intensity of a point in the $n$-th field having horizontal and vertical coordinates $x$ and $y$ respectively. Also, let $a$ and $b$ be the maximum allowable horizontal and vertical displacements respectively of pels between two fields under consideration.

**Definition 1.** We say that a pel $(\hat{x}, \hat{y}, j)$ in the $j$-th field *corresponds* to a reference pel $(x, y, i)$ in the $i$-th field if

$$| I(x, y, i) - I(\hat{x}, \hat{y}, j) | = \min_{\substack{x-a \le x' \le x+a \\ y-b \le y' \le y+b}} | I(x, y, i) - I(x', y', j) |,$$

i.e., the pel $(\hat{x}, \hat{y}, j)$ matches the intensity of the reference pel more closely than any other pel in the neighboring $(2a + 1) \times (2b + 1)$ rectangle. Then displacement D of $(x, y, i)$ is defined as

$$\mathbf{D} = \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}.$$

It is important to realize that the above definition relies on the assumed picture model. For example, it is easy to construct an object, violating the above model, and its displaced version such that the displacement determined

using the above definition will significantly differ from the actual displacement. However, in the applications of interest, the images largely conform to the selected model. We, therefore, adopt it for our algorithm.

In the following discussion, we will somewhat condense the notation and make a number of new definitions. Let

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \hat{\mathbf{x}} = \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}.$$

Further, we denote the intensity error (also referred to as displaced field difference) as

$$DFD(\mathbf{x},\mathbf{D}) \equiv I(\mathbf{x}, i) - I(\mathbf{x} - \mathbf{D}, j).$$

The minimization required in Definition 1 can be carried out by finding the local minimum of some positive error function such as $|\,DFD\,|$, $DFD^2$, etc., in the vicinity of $(\mathbf{x}, j)$. This approach will generally not yield the global minimum in the required region unless additional restrictions on the image model are imposed. Although these restrictions cannot be guaranteed, the problem is solved by exploiting the correlation between the displacements of neighboring pels. The details of this will be given later. The local optimization is implemented here using a recursive steepest descent algorithm. For $DFD^2$, the recursion is

$$\begin{aligned} \mathbf{D}_n &= \mathbf{D}_{n-1} - (\epsilon/2) \cdot \nabla_{\mathbf{D}} [DFD(\mathbf{x}, \mathbf{D}_{n-1})]^2 \\ &= \mathbf{D}_{n-1} - \epsilon \cdot DFD(\mathbf{x}, \mathbf{D}_{n-1}) \cdot \nabla_{\mathbf{D}} DFD(\mathbf{x}, \mathbf{D}_{n-1}), \end{aligned} \tag{1}$$

whereas for $|\,DFD\,|$,

$$\begin{aligned} \mathbf{D}_n &= \mathbf{D}_{n-1} - \epsilon \cdot \nabla_{\mathbf{D}} |\,DFD(\mathbf{x}, \mathbf{D}_{n-1})\,| \\ &= \mathbf{D}_{n-1} - \epsilon \cdot \operatorname{sign}[DFD(\mathbf{x}, \mathbf{D}_{n-1})] \cdot \nabla_{\mathbf{D}} DFD(\mathbf{x}, \mathbf{D}_{n-1}), \end{aligned} \tag{2}$$

where

$$\operatorname{sign}\alpha = \begin{cases} 1 & \text{if } \alpha > 0 \\ 0 & \text{if } \alpha = 0 \\ -1 & \text{if } \alpha < 0 \end{cases},$$

and $\nabla_{\mathbf{D}}$ is the gradient with respect to displacement $\mathbf{D}$. The gradient can be evaluated from the definition of $DFD$ as

$$\nabla_{\mathbf{D}} DFD(\mathbf{x}, \mathbf{D}_{n-1}) = \nabla I(\mathbf{x} - \mathbf{D}_{n-1}, j). \tag{3}$$

The positive scalar constant $\epsilon$ is chosen emperically so as to achieve fast convergence, yet avoid oscillations.

Notice that Equation (2) is a simplified version of Equation (1). An alternative simplification yields

$$\mathbf{D}_n = \mathbf{D}_{n-1} - \epsilon \cdot DFD(\mathbf{x}, \mathbf{D}_{n-1}) \cdot \mathrm{sign}[\nabla_{\mathbf{D}} DFD(\mathbf{x}, \mathbf{D}_{n-1})]. \tag{4}$$

We found that Equation (4) gives better subjective results than Equation (2) and therefore chose only it for further investigations.

Application of the steepest descent algorithm to the image motion estimation was first proposed by Netravali and Robbins [5] who also showed that the algorithm converges. However, the speed of convergence is low, while in real-time video processing we can rarely afford more than one iteration per pel. This results in errors and incoherency of the displacement estimates which subsequently produce rather objectionable distortions in the interpolated fields.

To alleviate the problem, the complete estimated displacement field is smoothed using a median filter. A median filter acting on a pel picks the values of this and the surrounding pels, sort them in an ascending (or descending) order, and replace the original pel by the median of the sorted array. We chose such smoothing procedure over linear low-pass filtering which may modify initially correct estimates and/or destroy sharp moving area boundaries.

Finally, once the displacement field from field $i$ to field $j$ is available, the interpolation of pels in the $k$-th ($j \leq k \leq i$) field is accomplished by linearly weighting and summing appropriately displaced pels in the $i$-th and $j$-th fields.

## 2.3. Performance Evaluation.

The motion compensated interpolation algorithm described here has been simulated implemented using BNR/INRS Image processing facilities. The test sequences included head-and-shoulder views of a conference participant undergoing various degrees of motion. The quality of the images also varied from broadcast high fidelity to the reduced quality produced by a 50 Kb/s coder. The high quality luminance sequences were obtained by sampling the demodulated NTSC signal at $4f_{sc}$ (subcarrier frequency) and quantizing each sample to 8 bits. Only a window about 1/7 the size of a conventional TV screen has been examined which combined with the above sampling rate gave $256 \times 106$ pels per field at 60 fields per second. Such high-quality sequences were used to isolate the interpolation problems from those caused by coding. In the coding environment, on the other hand, the processed sequences presented to the field interpolator were subsampled to $2f_{sc}$ for the luminance and $0.25f_{sc}$ for the chrominance. The resulting luminance field consisted of $128 \times 106$ pels. In this context, the intensity values have also been rather coarsely quantized, although the actual quantizer step varied adaptively with image statistics. All the tests were evaluated on a purely subjective basis, since no meaningful quantitative measures were found to describe the amount of artifacts and perceptual impairments present in the picture.

In the cases of interest, the field subsampling ratio ranges from 4:1 to 10:1, although higher ratios are sometimes required in certain modes of the coder operation. The algorithm parameters were optimized for the subsampling ratios of 4:1 and 6:1, but also performed well, albeit suboptimally, for higher ratios. These parameters were found to be the following.

1) The maximum horizontal and vertical displacements $a$ and $b$, as in Definition 1, are restricted to 18 pels and 9 lines for $256 \times 106$ images respectively, and 9 pels and 9 lines for $128 \times 106$ images respectively.

ii) The motion detection threshold (the limit on the interframe difference beyond which a pel is classified as moving) gave best results when set to 5 units on a 256 level (8 bits) scale. When the quantizer step of the processed pictures is increased above 5, however, this threshold is also increased to the size of one quantizer step.

iii) The convergence coefficient $\epsilon$ in the recursion (4) was selected among power of 2 values to be 1/128. The optimization was done for images sampled at $4f_c$ and temporally subsampled at 4:1 and 6:1. Further spatial and temporal subsampling may call for an increase in this value.

The analyzed algorithm generally produced very favorable results. When it was applied to high quality black-and-white sequences temporally subsampled at the ratio of 4:1, the reconstructed images of the speaker retained high resolution and perceptually low distortion. Only in the A-B comparison tests with the original sequence some loss of lip motion was noticed. On the other hand, when compared to linearly interpolated sequences, a better definition and higher resolution were apparent in the new process. At higher subsampling ratios, artifacts began to appear. Usually, the artifacts were due to slow updating of the displacement vectors. As a result, unrelated parts of the image underwent the same motion. For example, in one instance, the subject's collar moved in the same direction as his head, whereas in the original, the direction was different. However, jerkiness in the motion was not present, and the artifacts were tolerable up to subsampling ratios of over 10:1, after which they became objectionable. In subjective testing, these artifacts were unanimously preferred to those produced by linear interpolation. For quantitative comparison, the mean-squared error averaged over the complete image is given in Figures 2.4–2.6. It is thought, however, that the mean-square error measure of the algorithm's performance is less indicative of picture quality than the subjective testing.

It is interesting to notice that when applied to lower resolution images,

as is the case in the given coder, the interpolation artifacts appeared masked. In general, it was found that the motion rendition was as good in the low resolution environment as is it was in high resolution environment. The important conclusion is, therefore, that the motion-compensated interpolation algorithm considered in this report is robust with respect to the operating environment, as predictable results were obtained for different spatial and temporal subsampling ratios and various degrees of quantization.

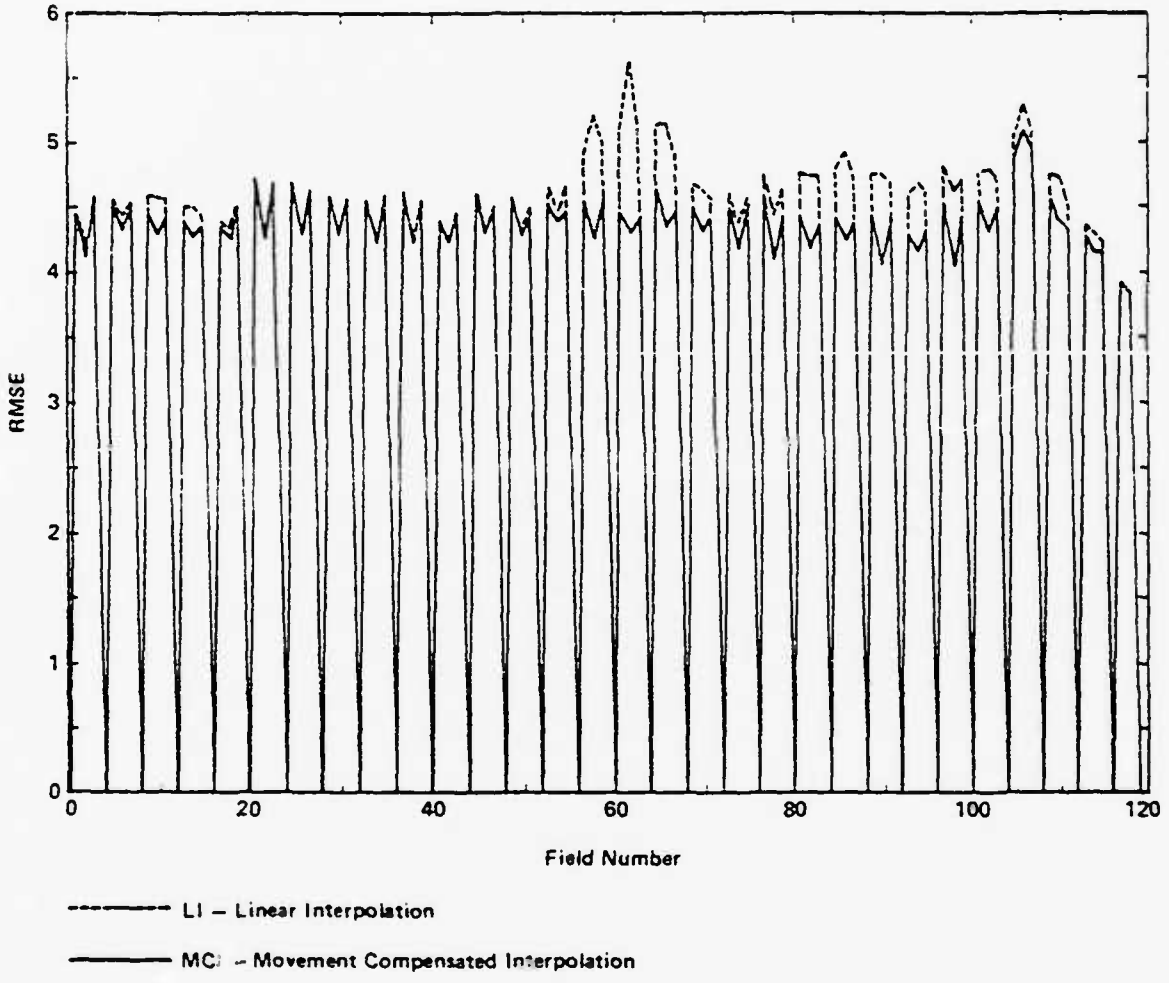



Figure 2.4: RMS Error Generated by 4:1 Field Interpolation

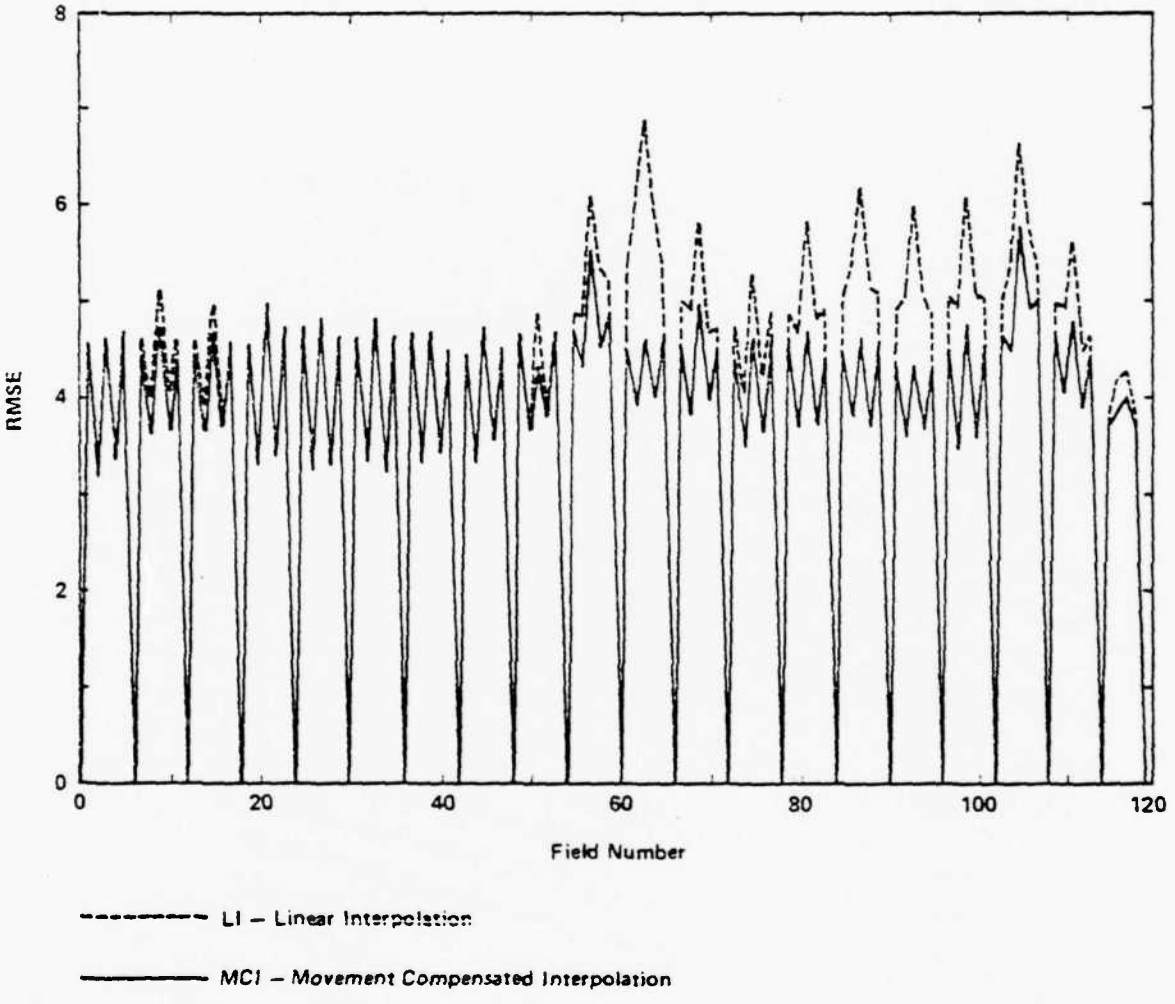



Figure 2.5: RMS Error Generated by 6:1 Field Interpolation

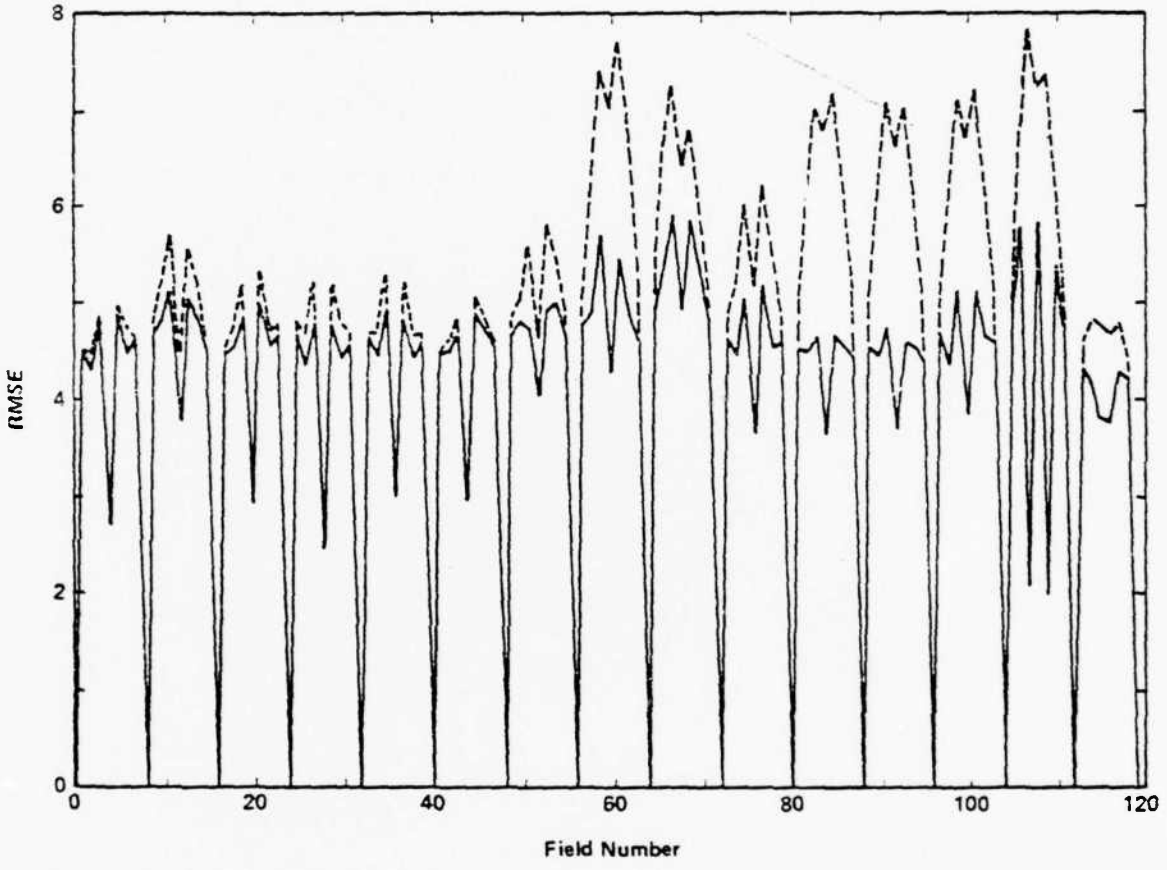



Figure 2.6: RMS Error Generated by 8:1 Field Interpolation

# 3. The Multi-Mode Coder.

In this chapter we describe the funcional blocks of the coder indicated in Figure 1.1. Most of the coder components are only reviewed here; for the details the reader is referred to [2]. Spatial sampling process is addressed in greater detail.

## 3.1. Demodulation and Signal Conditioning.

At the front end of the transmitter is the demodulation circuit which accepts the composite color NTSC video signal sampled at $4f_{sc}$ and outputs three color components: one luminance and two chrominance. This is achieved using 2-dimensional spatial filtering as shown in Figure 3.1. The comb and the band-pass filters used in the demodulator both have a finite impulse response of the form

$$h_{comb}(n) = (-1, 2, -1)/4$$

and

$$h_{bandpass}(n) = (-1, 0, 8, 0, -15, 0, 20, 0, -15, 0, 8, 0, -1)/64$$

respectively.

Following the demodulator is the noise reducer. The configuration of this unit, shown in Figure 3.2, is similar to that of a differential coder with a quantizer replaced by a non-linear element and the output taken where the reconstructed signal is produced. The non-linearity characteristics are shown in Figure 3.3. The overall effect is that of suppressing noise in the stationary areas. A more detailed discussion of the noise reduction process is found in [2].

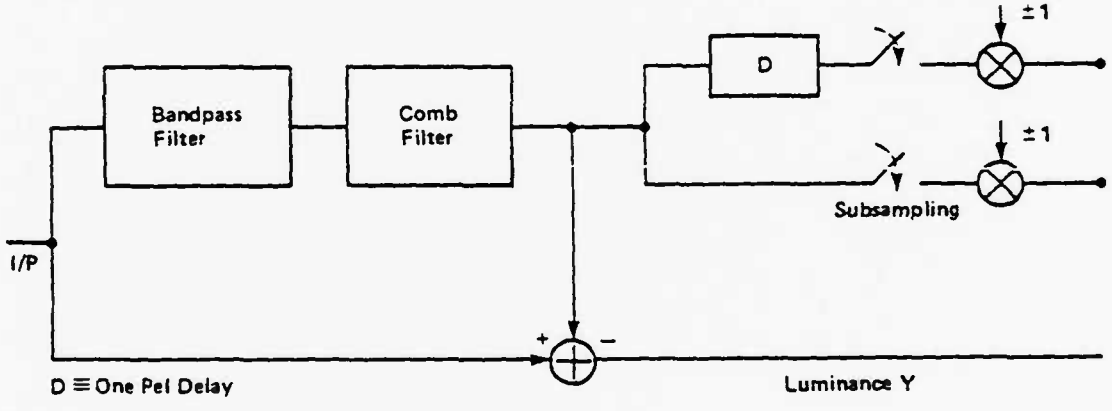



Figure 3.1: Digital Demodulation of the Composite NTSC Video Signal Sampled at $4*f_{SC}$ (14.3MHz)

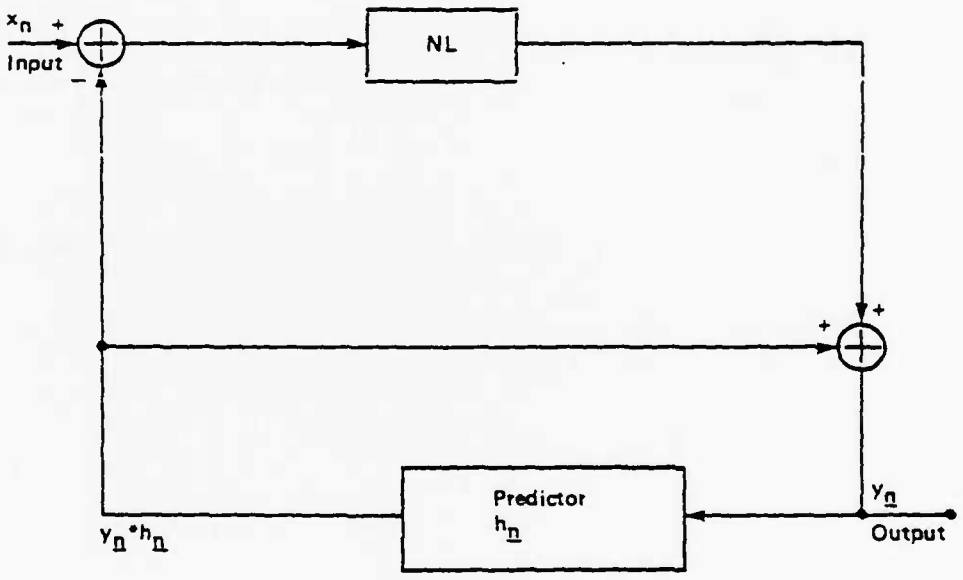



Figure 3.2: Block Diagram of the Digital Noise Reducer

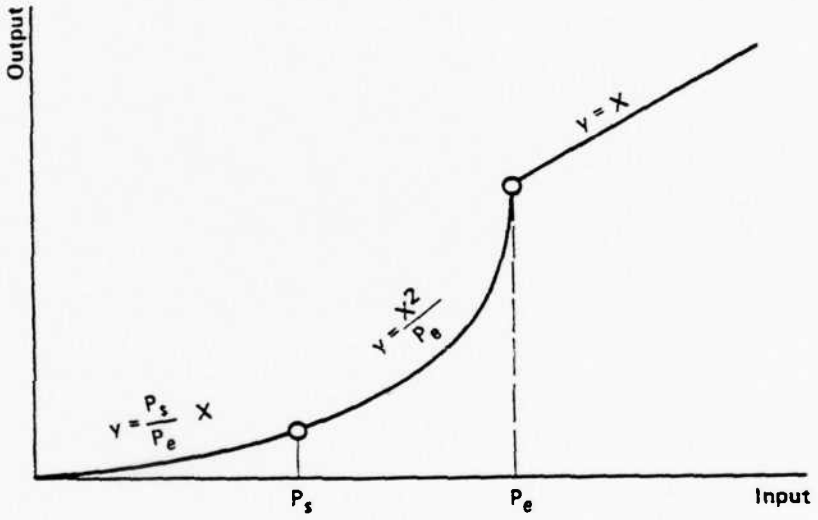



Figure 3.3: Nonlinearity Used for Noise Reduction

## 3.2. Spatial Subsampling and Interpolation.

Digital processing and transmission of analog video signals requires the process of sampling to be performed on the signals. The NTSC video format provides temporal sampling of continuously moving images to 60 fields/sec, and vertical sampling to 525 lines per frame, or $262\frac{1}{2}$ lines per field. The lines in odd fields are vertically offset by half a line with respect to even fields, thus producing interlaced images which effectively double the vertical sampling frequency. Finally, for digital transmission, the continuous NTSC signal is sampled at $4f_{sc}$, $f_{sc} \approx 3.58$MHz.

In order to achieve 50 kb/s transmission rate, the number of samples (pels) to be processed per unit time must be drastically reduced. This is accomplished by restricting the view to a window of size 256 pels $\times$ 106 lines, and then performing spatial and temporal subsampling. Temporal subsampling and interpolation has been discussed in the previous chapter. In this section, only spatial subsampling is discussed.

The sampling process may introduce aliasing (distortion in the spectrum) unless the input signal is bandlimited to half the sampling frequency. In the geven application, aliasing is particularly undesirable, since it not only degrages the picture quality, but also affects the robustness of the motion estimates. Therefore, the input signal, initially sampled to $4f_{sc}$, is bandlimited to about 2MHz prior to further subsampling. This operation does not introduce significant distortions, since the frequencies being cut off contain little video information. The bandlimiting filter has the impulse response

$$h(n) = (1, 0, -6, 0, 15, 0, 44, 0, 15, 0, -6, 0, 1)/64.$$

The bandwidth of 2MHz is still greater than $0.5f_{sc}$, $f_{sc}$ being the target sampling frequency. However, further advantages can be taken of the two-dimensional properties of the video signal. Subsampling the above signal

at a ratio 2:1 by simply dropping every second column produces the two-dimensional spectrum shown in Figure 3.5 [6], where $f_1$ and $f_2$ are the horizontal and the vertical frequencies respectively. The above sampling pattern is called orthogonally aligned, and is shown in Figure 3.4.

Further orthogonally aligned subsampling by a factor of two (Figure 3.6) will introduce aliasing as shown in Figure 3.7. In applications where aliasing can be tolerated, the orthogonally aligned pattern has the advantage that a one-dimensional horizontal filtering can be used to reconstruct the signal. However, in our application, aliasing is particularly undesirable, because in addition to deteriorating the picture quality, it causes errors in the motion estimator used in motion compensated coding and interpolation. As a result, the transmission rate is unnecessarily increased, and the distortions generated by the temporal field interpolation reach an unacceptable level.

An alternative sampling pattern is line quincunx shown in Figure 3.8. The corresponding spectrum is in Figure 3.9 [6]. The high-frequency components do not overlap with the baseband, and can be removed by a filter. The reconstruction filter is two-dimensional with finite impulse response

$$h(n) = \frac{1}{32}\begin{bmatrix} 0 & -3 & 0 & 8 & 0 & -3 & 0 \\ 1 & 0 & 15 & 0 & 15 & 0 & 1 \\ 0 & -3 & 0 & 8 & 0 & -3 & 0 \end{bmatrix}.$$

Although the above discussion concerns only the luminance part of the NTSC signal, the results also apply to the chrominance components, which initially need to be sampled only to $0.5f_s$ each. Following the 4:1 spatial subsampling procedure, the pattern of remaining samples is shown in multiplexed form in Figure 3.8. Combination of windowing and spatial subsampling alone affords a reduction of the bit rate by a factor of 28 as compared to the initial digitized signals.

There are several places in the signal path where the reconstructing interpolation filters can be inserted. The most obvious one is the output

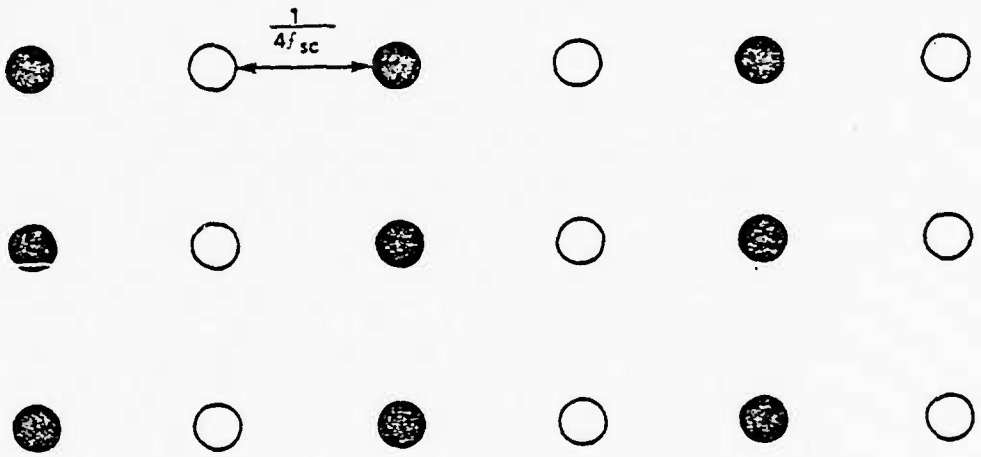



Figure 3.4: Orthogonally Aligned 2:1 Subsampling

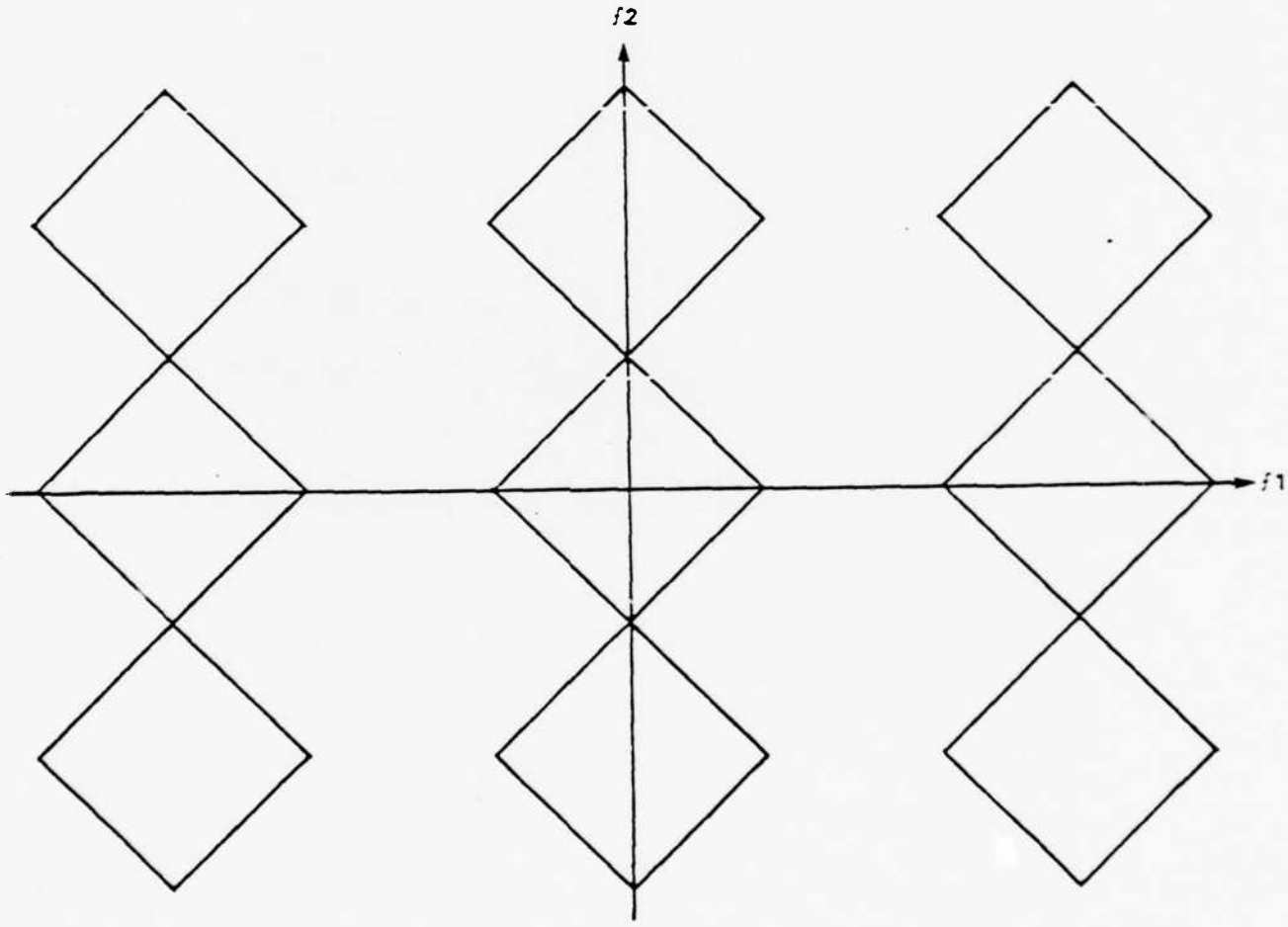



Figure 3.5: Two-dimensional Spectrum of the Orthogonally Aligned 2:1 Subsampling

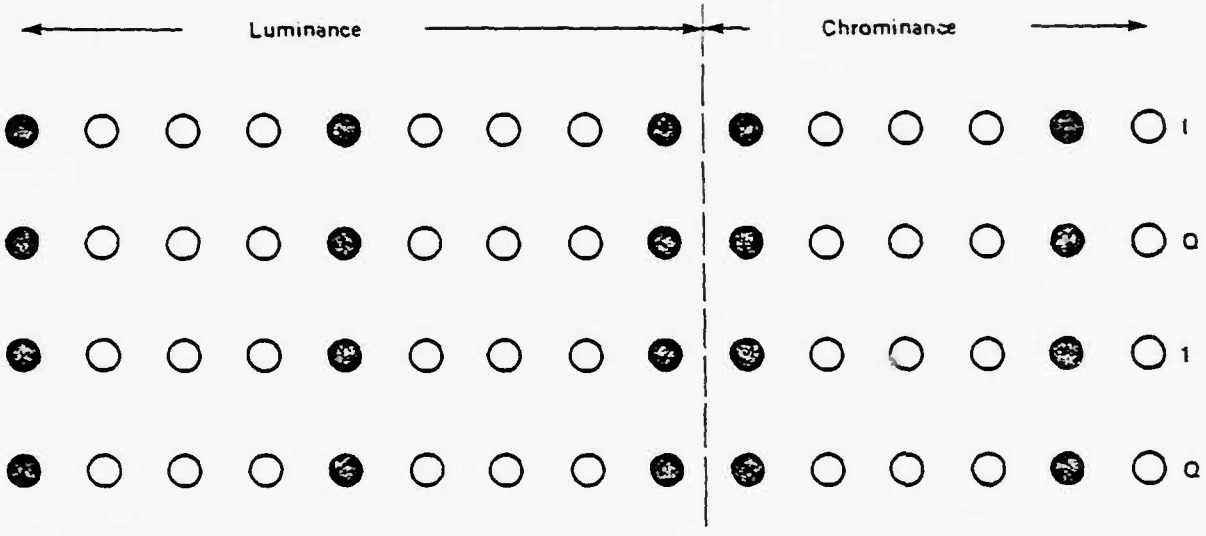



Figure 3.6: Orthogonally Alligned
4:1 Subsampling of the Multiplexed Colour Components

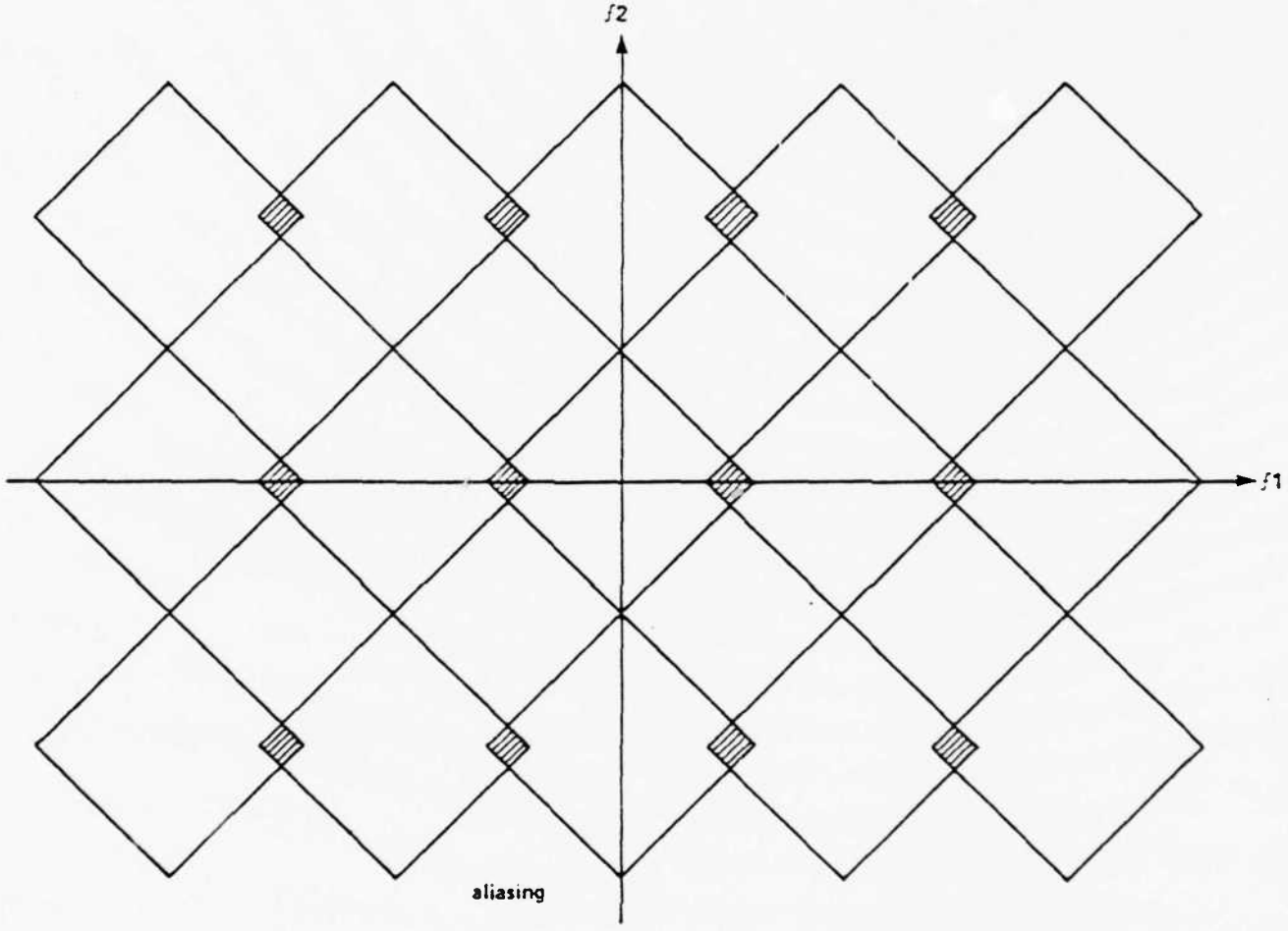



Figure 3.7: Spectrum of the Orthogonally Aligned 4:1 Subsampling

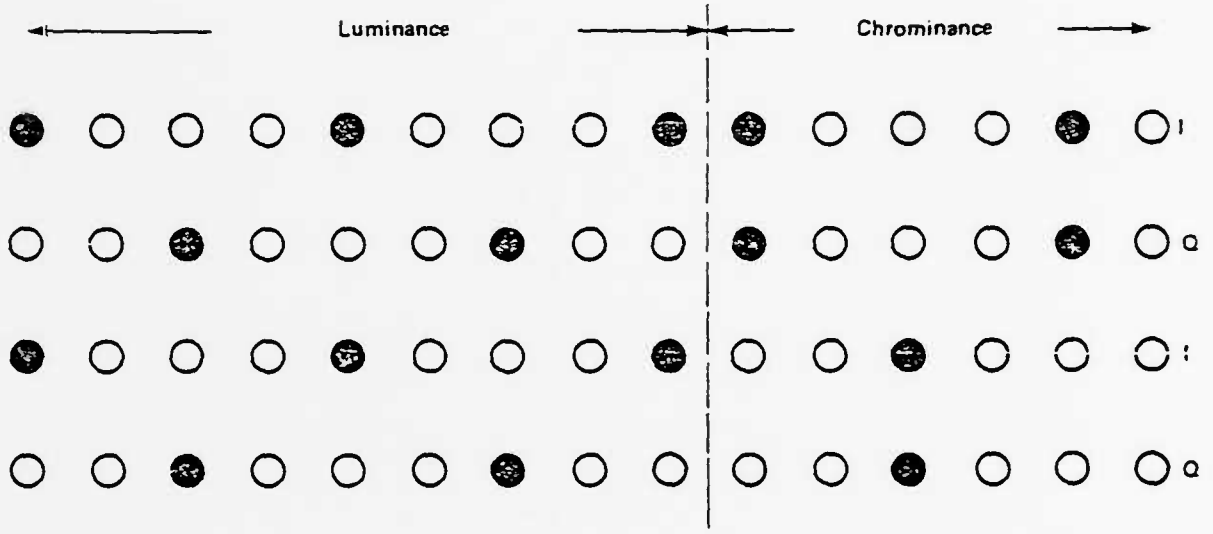



Figure 3.8: Line Quincunx
4:1 Subsampling of the Multiplexed Colour Components

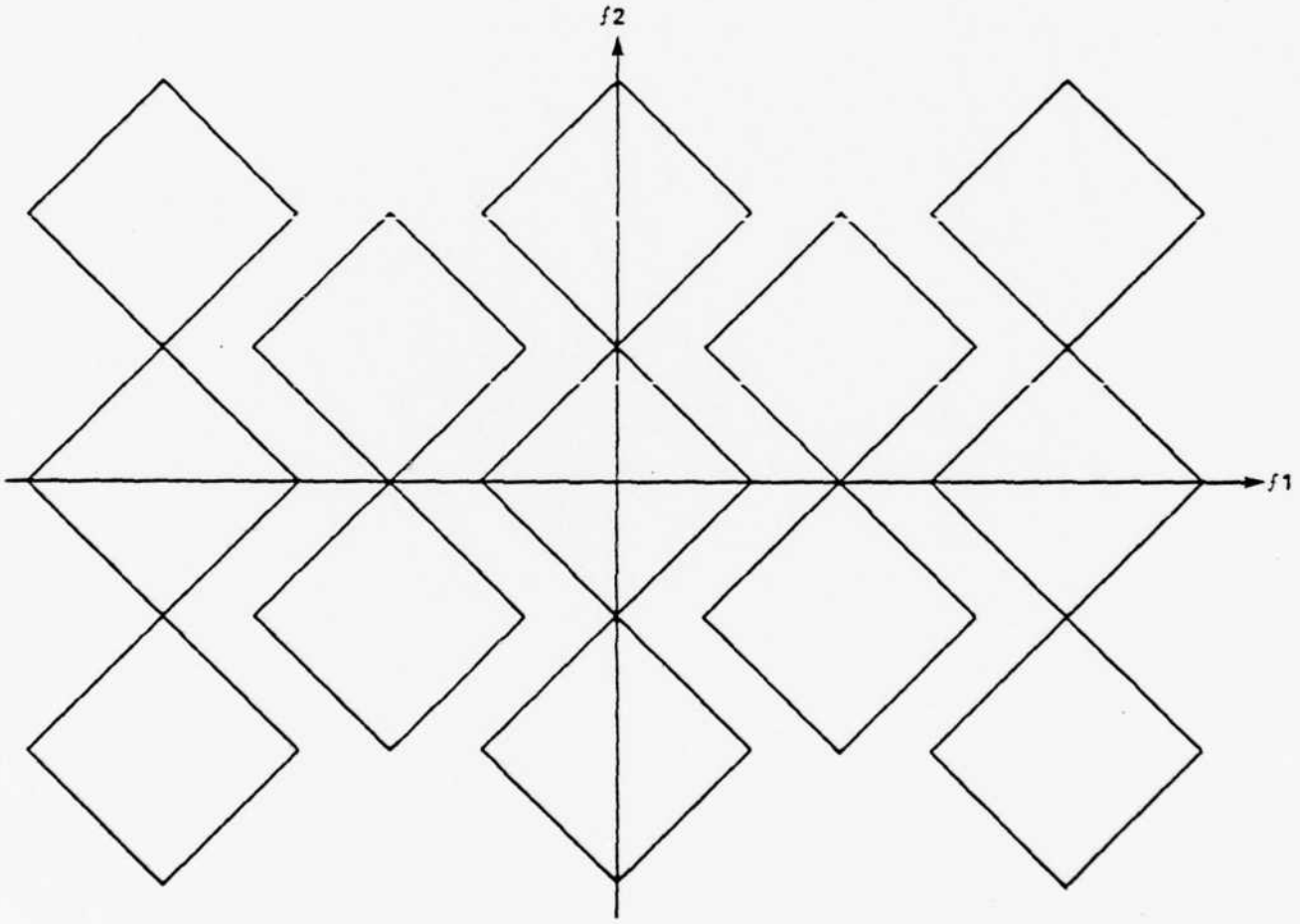



Figure 3.9: Spectrum of the Line Quincunx 4:1 Subsampling

of the receiver, just before the image is displayed. However, in the context of movement compensated coding, it is preferable to perform interpolation earlier. The robustness of the prediction required in such a coding scheme depends on the accuracy of the reproduction of previous fields. This accuracy is enhanced if the interpolation is done within the DPCM feedback loop, as shown in Figure 3.10. This way, the image presented to the displacement estimator is of higher resolution; hence the prediction can be formed more accurately.

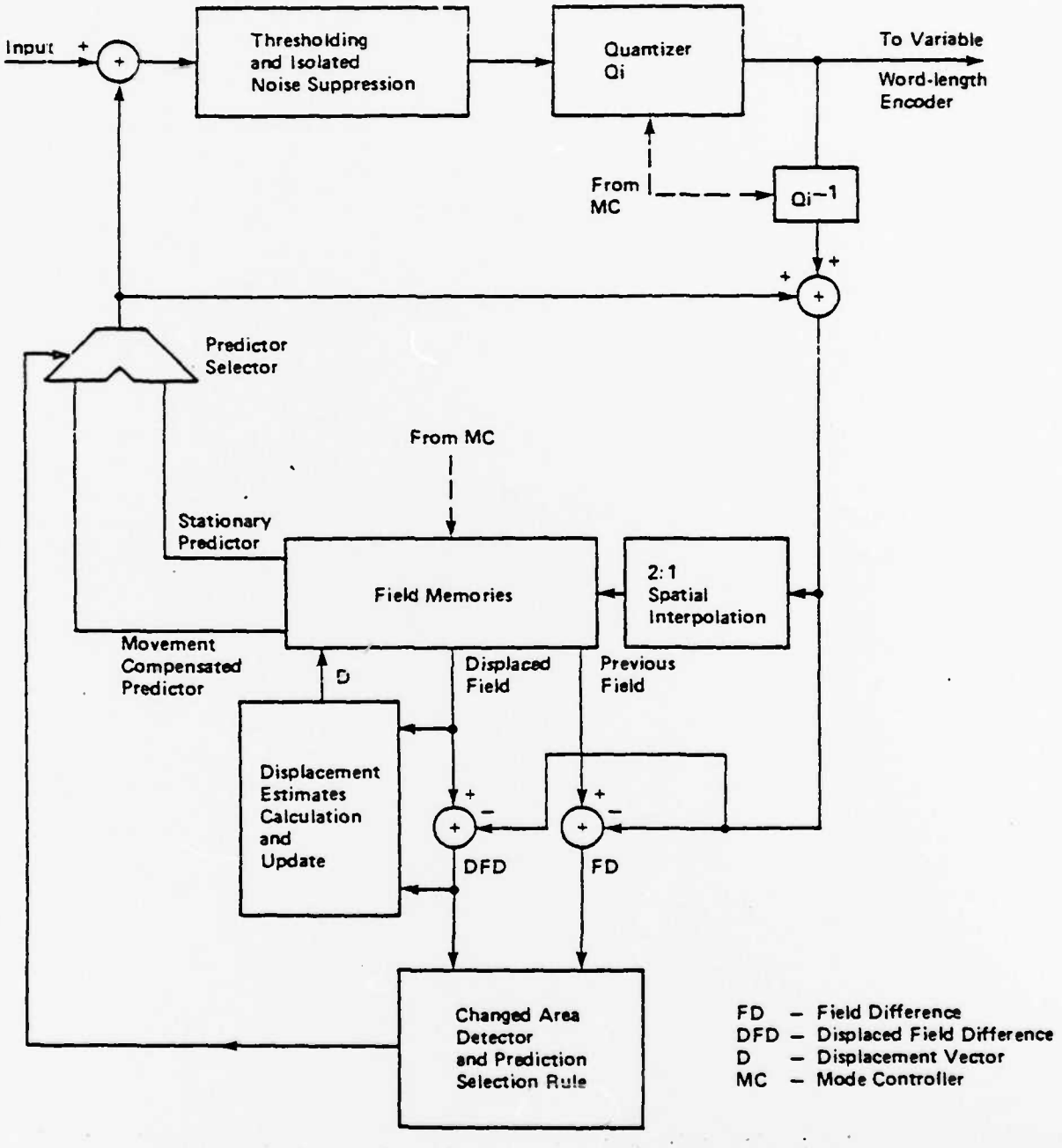



Figure 3.10: Block Diagram of the Movement Compensated Interframe Video Coder

## 3.3. Motion Compensated Coder.

The block diagram of the complete motion compensated coder is shown in Figure 3.10. The most important part of the motion compensated coder is the motion estimator. Details of this algorithm are given in [5]. Although similar to the motion estimating algorithm used for motion compensated field interpolation, the current algorithm is different in several ways. Here, we will concentrate only on these differences; for the basics of the algorithm operation, the reader is referred back to the section on the motion compensated interpolation.

The motion estimation algorithm operates on images sampled at a rate of $1/_{sc}$. Therefore, the motion updates generated according to Eq. (4) (Chapter 2) are produced only at half the rate of $2/_{sc}$ used in field interpolation. This obviously affects the accuracy of the estimates. To alleviate the problem, the accuracy of the updates is increased. This is accomplished by interpolating the previous field to $2/_{sc}$ within the feedback loop, immediately after it has been coded and reconstructed. Hence, even though the updates are still generated at a rate of $1/_{sc}$, their accuracy is enhanced due to the more precise computation of $DFD$ and the gradient (see Eq. (4)). The exact placement of the 2 : 1 spatial interpolator is shown in Figure 3.10.

The selection rule between the zero and non-zero displacements is based on two pels located above the one being processed, as shown in Figure 3.11. The selection rule is summarized in the following equation:

$$\begin{aligned} &\mathrm{D}(\mathbf{x},i) = 0 \quad \text{if } |FD_1| + |FD_2| < |DFD_1| + |DFD_2| \\ &\mathrm{D}(\mathbf{x},i) \neq 0 \quad \text{otherwise} \end{aligned}$$

where $FD_k$ is the previous frame difference and $DFD_k$ is the displaced frame difference of the pel $k$, as shown in Figure 3.11. This rule is different from the one used in field interpolation where the selection depends on the prediction error of the previous pel in the current line.

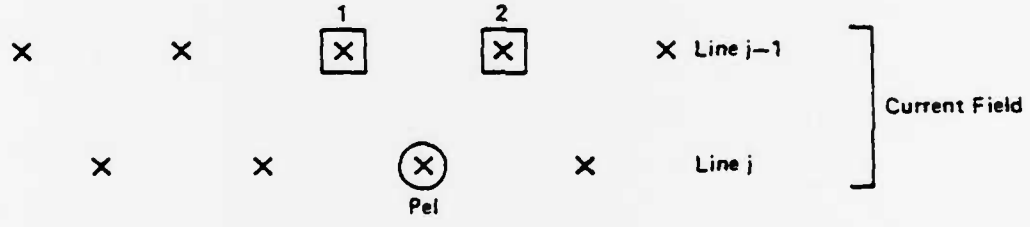



Figure 3.11: Prediction Selection Rule for
Luminance Part of the Multiplexed Signal

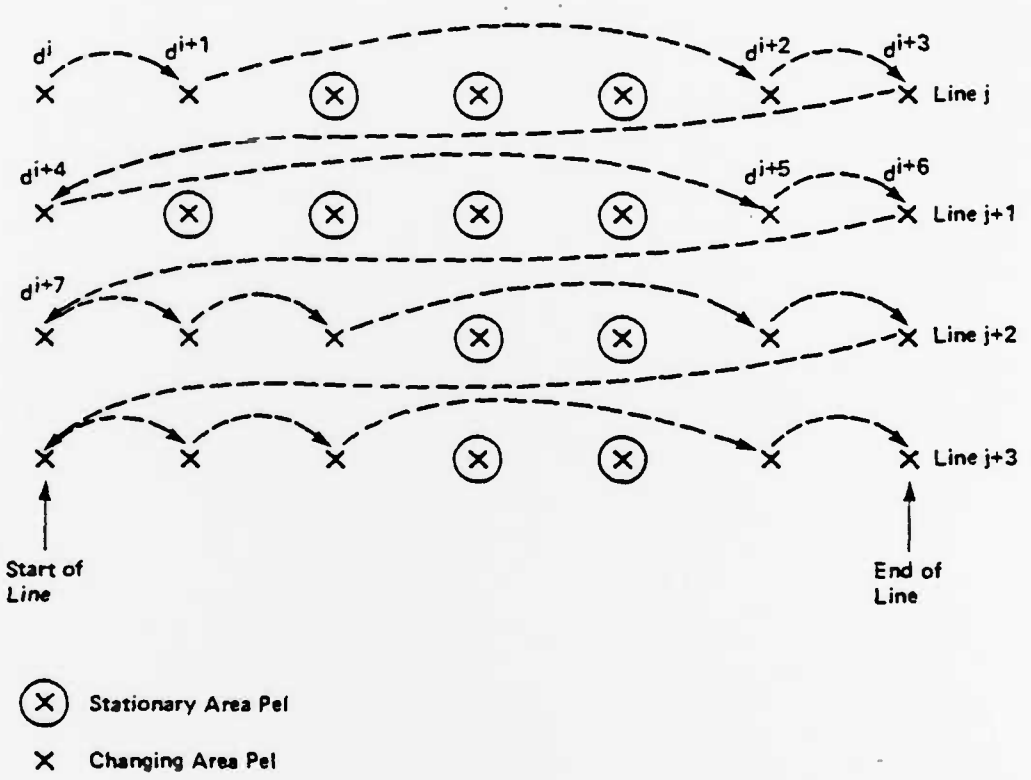



Figure 3.12: Illustration of the Displacement Estimate Updating

The order in which the motion estimates are updated, and patterns along which previously computed displacement estimates propagate in the coder are illusstrated in Figure 3.12. The pels are processed from left to right in every line. As a result, propagation of the displacement estimates in the vertical direction occurs only on the transition from the end of the current line to the beginning of the next.

The motion estimator used in the coder has tighter time constraints than the one used in the field interpolator. Presence of the feedback loop in the coder places a restriction on the maximum time it takes the signal to propagate around the loop for proper operation. This, in turn, restricts the processing time allowed for a single pel. In contrast, the field interpolator contains no feedback and is therefore unconditionally stable. Hence, the use of pipelining techniques allows any amount of processing to be performed, as long as the delay is acceptable to the users.

| Level # | Signed binary representation | Variable word length representation | Word length |
|---|---|---|---|
| -15 | 1 0 1 1 1 1 | 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 | 17 |
| -14 | 1 0 1 1 1 0 | 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 | 16 |
| -13 | 1 0 1 1 0 1 | 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 | 15 |
| -12 | 1 0 1 1 0 0 | 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 | 14 |
| -11 | 1 0 1 0 1 1 | 1 1 0 0 0 0 0 0 0 0 0 0 0 1 | 13 |
| -10 | 1 0 1 0 1 0 | 1 1 0 0 0 0 0 0 0 0 0 0 1 | 12 |
| -9 | 1 0 1 0 0 1 | 1 1 0 0 0 0 0 0 0 0 0 1 | 11 |
| -8 | 1 0 1 0 0 0 | 1 1 0 0 0 0 0 0 0 0 1 | 10 |
| -7 | 1 0 0 1 1 1 | 1 1 0 0 0 0 0 0 0 1 | 9 |
| -6 | 1 0 0 1 1 0 | 1 1 0 0 0 0 0 0 1 | 8 |
| -5 | 1 0 0 1 0 1 | 1 1 0 0 0 0 0 1 | 7 |
| -4 | 1 0 0 1 0 0 | 1 1 0 0 0 1 | 6 |
| -3 | 1 0 0 0 1 1 | 1 1 0 0 1 | 5 |
| -2 | 1 0 0 0 1 0 | 1 1 0 1 | 4 |
| -1 | 1 0 0 0 0 1 | 1 1 1 | 3 |
| 0 | 0 0 0 0 0 0 | 0 | 1 |
| 1 | 0 0 0 0 0 1 | 1 0 1 | 3 |
| 2 | 0 0 0 0 1 0 | 1 0 0 1 | 4 |
| 3 | 0 0 0 0 1 1 | 1 0 0 0 1 | 5 |
| 4 | 0 0 0 1 0 0 | 1 0 0 0 0 1 | 6 |
| 5 | 0 0 0 1 0 1 | 1 0 0 0 0 0 1 | 7 |
| 6 | 0 0 0 1 1 0 | 1 0 0 0 0 0 0 1 | 8 |
| 7 | 0 0 0 1 1 1 | 1 0 0 0 0 0 0 0 1 | 9 |
| 8 | 0 0 1 0 0 0 | 1 0 0 0 0 0 0 0 0 1 | 10 |
| 9 | 0 0 1 0 0 1 | 1 0 0 0 0 0 0 0 0 0 1 | 11 |
| 10 | 0 0 1 0 1 0 | 1 0 0 0 0 0 0 0 0 0 0 1 | 12 |
| 11 | 0 0 1 0 1 1 | 1 0 0 0 0 0 0 0 0 0 0 0 1 | 13 |
| 12 | 0 0 1 1 0 0 | 1 0 0 0 0 0 0 0 0 0 0 0 0 1 | 14 |
| 13 | 0 0 1 1 0 1 | 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 | 15 |
| 14 | 0 0 1 1 1 0 | 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 | 16 |
| 15 | 0 0 1 1 1 1 | 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 | 17 |

Table 1: Variable Word Length Codes Used in the Coder

## 3.4. Other Bit-Rate Reduction Techniques Employed.

To exploit fully the statistical redundancy in the video signal, various adaptive techniques producing variable bit rates have to be used. Consequently, an output buffer is necessary in order to convert the variable rate input data stream into a constant rate output. The modes of the coder which govern the adaption parameters are based on the percentage of the buffer occupancy. In addition to the coding algorithms above, the following bit rate reduction techniques are incorporated in the coder:

a) A *uniform quantizer* with an *adaptive step-size*. The step-size changes from 5 to 11 depending on the buffer occupancy, and hence, the mode of operation. As the buffer occupancy increases, quantization becomes coarser.

b) *Block coding* is used to identify blocks (rectangular areas) in which all prediction errors are zeros. Only one bit per block is required for this operation. If the block contains at least one non-zero prediction error, further encoding is necessary. If, on the other hand, the block is completely predictable (all prediction errors are zero), no additional information is needed.

c) *Variable-length codes* are used to encode the information within the non-zero blocks. Shorter codewords are assigned to more frequent events. A uniquely decodable set of codewords is required for this purpose. The set used in the present coder is given in Table 1.

The operation of the coder was simulated on the BNR/INRS VAX-780 computer and played back using the BNR proprietary digital video store (DVS) real-time display system. Several video sequences containing head-and-shoulder views have been used as input to the coder. The sequences contained varying amounts of motion in order to evaluate the coder performance.

The simulation showed that the performance of the above coder combined with the movement compensated field interpolation is dramatically improved over the previous algorithms. The jerkiness of the moving images has almost disappeared, and the motion rendition is greatly improved. The spatial resolution, although still low, was found to be better compared to the coder with linear field interpolation used before. Only large-motion segments which force the coder into high modes of operation produced objectionable distortions. However, such large motion is unlikely in video-conferencing applictions.

As an objective measure of performance, the buffer occupancy is determined for each sequence. This indicates which modes of operation the coder used in order to achieve the required transmission bit rate. Since higher modes of operation normally introduce degradation gracefully into the pictures, the buffer occupancy gives an indication of picture quality. The graphs of the buffer occupancy are plotted in Figures 13 - 15.

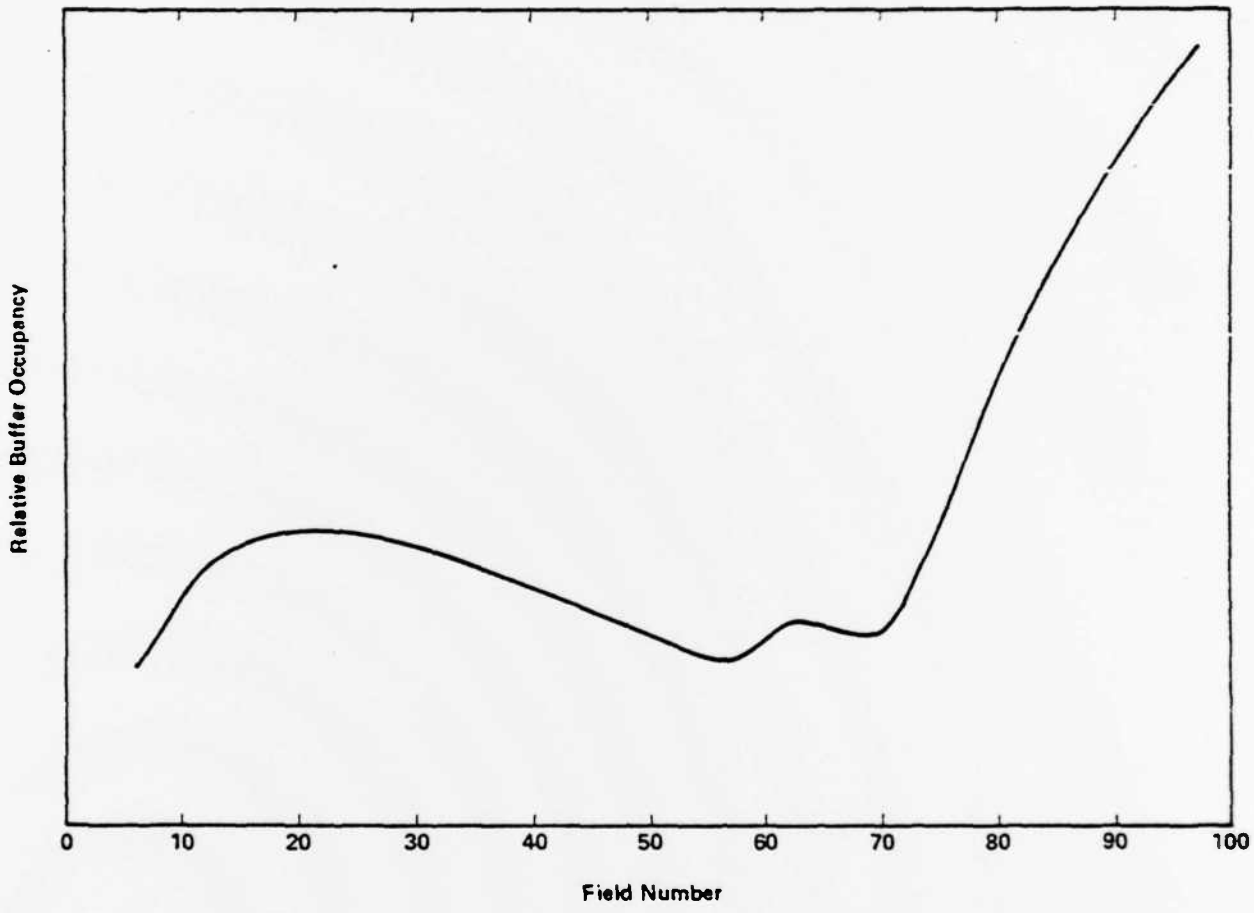



Figure 3.13: Buffer Occupancy for Sequence HARVEY

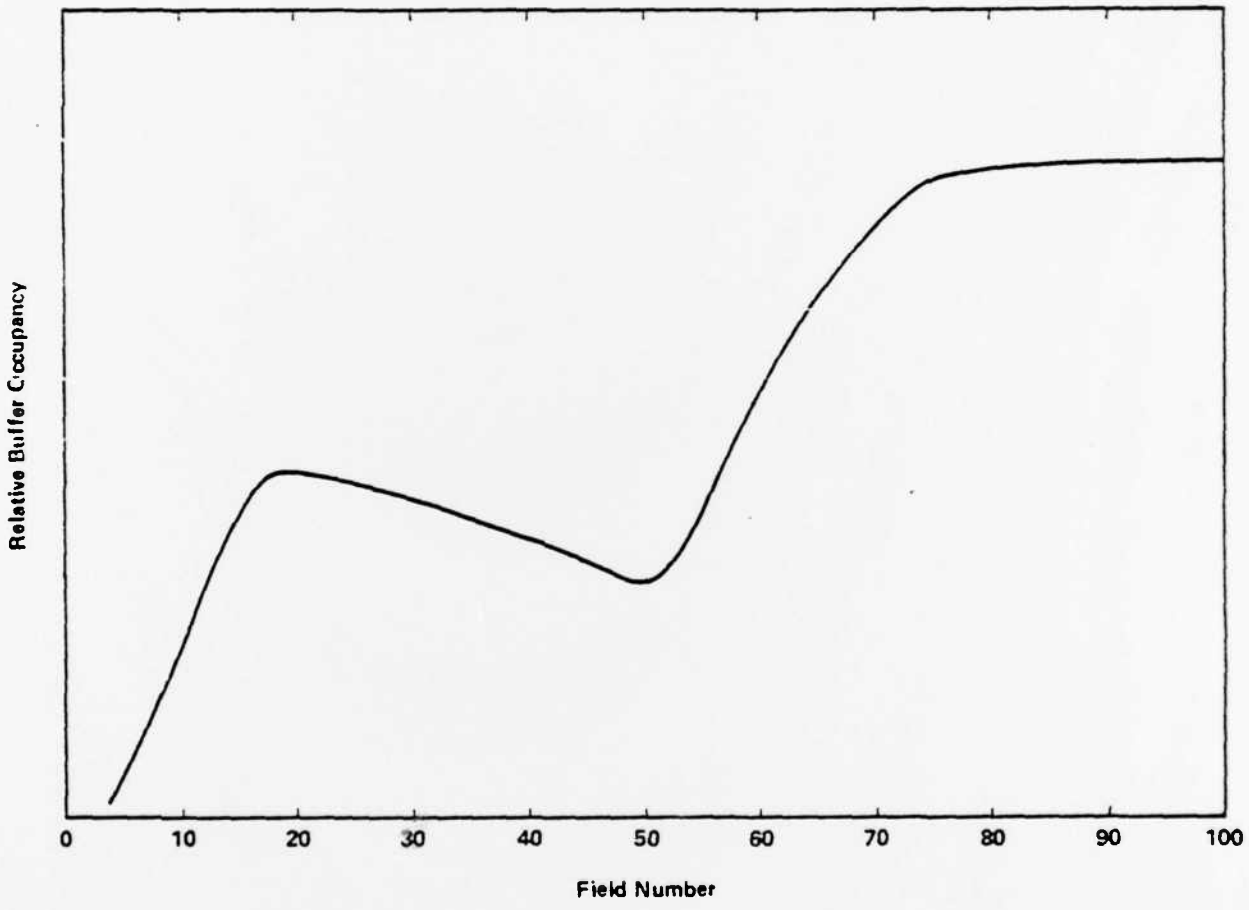



Figure 3.14: Buffer Occupancy for Sequence JACEK

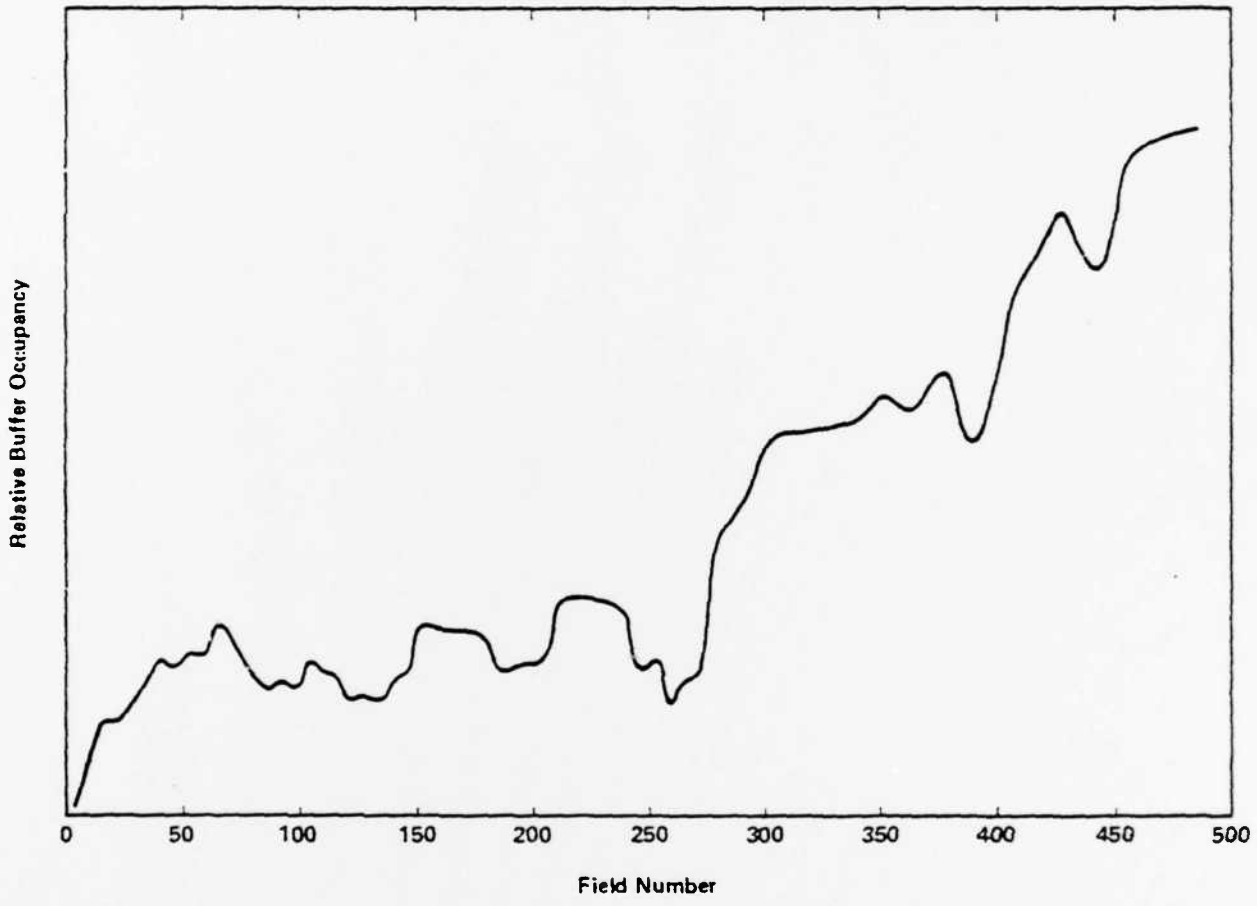



Figure 3.15. Buffer Occupancy for Sequence MARGARITA

## 4. Conclusions and Directions for Future Work.

In this study, 56 – 64 Kb/s movement compensated interframe coders for video conferencing applications have been presented. Particular attention was given to the spatial and temporal subsampling techniques. A considerable improvement in picture quality compared to the previously studied techniques was observed. This improvement is attributed mainly to the new movement compensated field interpolation algorithm. The encouraging results obtained using these new techniques prompts the interest in the following future developments.

Simplifications and the trade-offs of the current algorithm should be examined in order make it more amenable to the real-time implementation. In the present implementation, the motion compensated interpolation is a self-contained add-on package which works independently of the coding scheme employed. Therefore, even though the displacement field obtained during the motion compensated coding could be used for interpolation, a somewhat modified and enhanced motion estimation process is repeated here. In the future, however, the two procedures will be combined, and thus the total required computational complexity will be reduced.

Finally, the impact of transmission errors on the coder operation should be investigated. Suitable error correction and concealment techniques should be identified.

References.


1. C. W. Kelly III, "A Low Bandwidth Virtual Space Teleconferencing System," *Proceedings of 1982 GLOBECOM*, Miami, Florida, November 1982.

2. K. Cuffling and S. Sabri, "A Multi-Bit Rate Interprame Movement Compensated Multimode Coder for Video Conferencing," *Bell-Northern Research*, Final report prepared for DARPA under contract no. MDA903 – 81 – C – 0180.

3. E. Dubois "Movement Compensated Predictive Coding: Displacement Estimation and Coder Simulations," *Bell-Northern Research Technical Report* TM 32059, November 1979.

4. C. Cafforio and F. Rocca, "Detection and Tracking of Moving Objects in Television Images," in *Record of International Workshop on Image Processing: Real-Time Edge and Motion Detection/Estimation*, doc. Zich. CCETT: CTN/T/1/80, pp. 19.1-19.10, September 1979.

5. A. N. Netravali and J. D. Robbins, "Motion-Compensated Television Coding: Part I," *BSTJ*, Vol. 58, pp. 631-670, March 1979.

6. E. Dubois, B. Prasada, S. Sabri, *Image Sequence Coding*, Chapter 3, T. S. Huang ed., Springer-Verlag, New York 1981.